

## CLAIMS

What is claimed is:

1. A method for processing events from electronic architecture, the architecture of the type having one or more entities generating the events, comprising the steps of:

extracting the events from the architecture;  
separating the events according to the entities; and  
transforming the events to one or more text strings.

2. A method of claim 1, further comprising the step of filtering the events.

3. A method of claim 1, wherein the step of extracting the events comprises extracting chassis logs, wherein the step of separating the events comprises separating the chassis logs, and wherein the step of transforming events comprises transforming the chassis logs to text strings.

4. A method of claim 1, further comprising the step of coupling a getcc extraction tool to the architecture.

5. A method of claim 4, wherein the step of coupling comprises utilizing telnet.

6. A method of claim 1, the architecture being a server, and wherein the step of extracting events from the architecture comprises extracting events from the server.

7. A method of claim 1, wherein the step of transforming comprises converting a binary representation of the events to the text strings.

8. A method of claim 1, further comprising the step of analyzing the text strings and producing a human interpretable statement summarizing at least one of the events associated with the text strings.

9. A method of claim 1, wherein the entities comprises one or more of firmware, software, processors, architecture monitors, power monitors, cabinet monitors, and I/O drivers.

10. A method of claim 1, further comprising the step of controlling one or more steps of extracting, separating and transforming via one or more command line options.

11. A method of claim 10, further comprising controlling one or more steps of extracting, separating and transforming according to one or more configuration files.

12. A method of claim 10, wherein the step of controlling comprises inputting the command line options via a graphical user interface.

13. A method of claim 10, wherein the step of controlling comprises updating the command line options automatically from the architecture.

14. A method of claim 1, further comprising specifying one or more cells of the architecture, and extracting the events only from the one or more cells.

15. A method of claim 1, further comprising specifying one or more processors of the architecture, and extracting the events only from the one or more processors.

16. A method of claim 1, further comprising the step of saving a log file representative of the events.

17. A method of claim 1, further comprising the steps of transmitting the text strings to one or more analyzers associated with one or more entities and analyzing the text strings at the one or more analyzers.

18. A system for processing events from electronic architecture, the architecture of the type having one or more entities generating the events, comprising:  
an extraction tool for extracting the events from the architecture, separating the events according to the entities, and transforming the events to one or more text strings; and  
an interface for coupling the extraction tool to one or more of the architecture and a log file storing the events from the architecture.

19. A system of claim 18, wherein the entities comprise one or more of firmware, software, processors, architecture monitors, power monitors, cabinet monitors, and I/O drivers, and wherein the events comprise chassis logs from one or more of the firmware, software, processors, architecture monitors, power monitors, cabinet monitors, and I/O drivers.

20. A system of claim 18, further comprising one or more analyzers coupled to the extraction tool, the analyzers processing the text strings into one or more human interpretable statements summarizing at least one of the events associated with the text strings.